SymphonyAI
INDUSTRIAL

Plant Performance
WHITE PAPER

Optimal Maintenance Scheduler

Industrial process units consist of large numbers of assets working as a system to produce an output or a set of outputs. These outputs can consist of material (e.g., one or more products—intermediate or finished—that meet a set of specifications) or energy flows (e.g., heat or power generation).

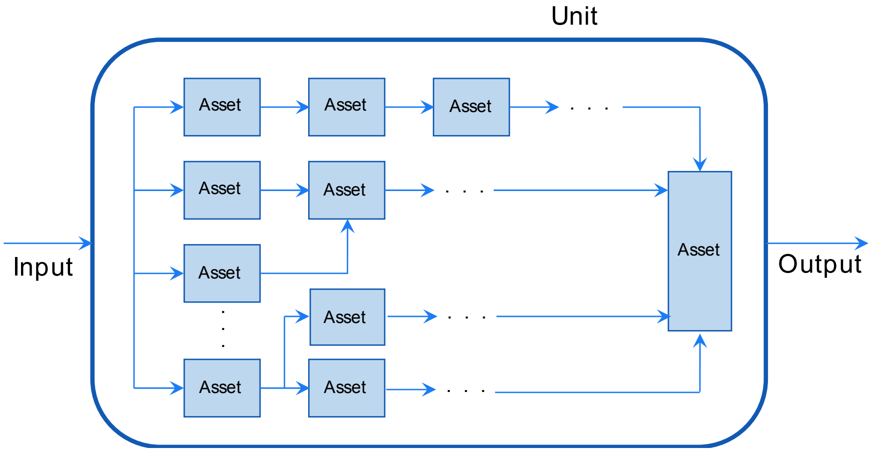

Figure 1: A unit in an industrial facility with several interconnected assets

Over time, the assets in the unit may degrade in performance. While they may not be in danger of experiencing unexpected downtime, their degradation may, and often does, result in reduced system output/throughput or increased system operational cost. They therefore need to undergo periodic maintenance to keep the system throughput at a desired level (or as high as possible). However, it is critical to optimally scheduling maintenance activities. Each maintenance activity incurs fixed costs (material and labour) as well as costs associated with reduced throughput during the maintenance period due to a reduction in available system capacity. Therefore, excessive maintenance can be as detrimental as too little maintenance.

SymphonyAI Industrial offers a generic maintenance optimisation framework — the Optimal Maintenance Scheduler (OMS) — that can be used to determine the optimal maintenance schedule for a wide variety of systems. It does this by minimising a user-specified customised cost objective function relevant to the unit. Since the objective function being optimised encapsulates performance at the unit- or plant-level (rather than asset-level), assets critical to the unit are prioritised for maintenance. In other words, the scheduler is more likely to recommend the maintenance of a partially degraded asset that has a greater effect on unit performance than that of a highly degraded asset that has little effect on unit performance.

The cost objective function may include several revenue/cost elements; some examples include:

1. Revenue associated with throughput (higher the throughput, higher the revenue)
2. Cost associated with throughput falling below a target level (or even exceeding a certain level)
3. Costs of inputs (consumables) such as fuel or power
4. Costs of maintenance activities

In the OMS, it is also possible to specify constraints on maintenance, e.g.:

1. Limiting the number of assets that can undergo maintenance at a given time due to resource limitations.
2. Scheduling asset maintenance activities so that they fall within specified periods (i.e., synchronising them with planned maintenance)

In what follows, we use the example of a refinery preheat train to demonstrate how the Optimal Maintenance Scheduler works. The preheat train in a refinery consists of several heat exchangers in series (occasionally in parallel) along with a furnace. The preheat train serves to heat the crude oil to a desired temperature before it enters the crude distillation unit. As heat exchangers in the train degrade, more fuel is required in the furnace to compensate for the reduced heating capacity of the heat exchangers. Optimal scheduling of heat exchanger maintenance is therefore important to keep operating costs low.

The OMS process

A schematic of the OMS process is shown in Figure 2. The main steps are:

1. Asset models are built for those assets in the unit that are to be considered in the maintenance scheduling problem. These asset models take as inputs the asset measurements and produce as outputs one or more measures of asset performance.
2. A system model is created for the whole unit. This model takes as inputs the outputs of the individual asset models as well as unit measurements, and produces as outputs one or more measures of unit performance (e.g., throughput, quality parameters, inputs required, etc.)
3. Forecast models are built for each of the assets using their respective asset models. These are used to forecast the asset performance out to a specified horizon (e.g., over the next 6 months)
4. The forecast models are used in the optimiser with inputs used to specify a cost objective function as well as maintenance constraints. The optimiser determines the optimal maintenance schedule.

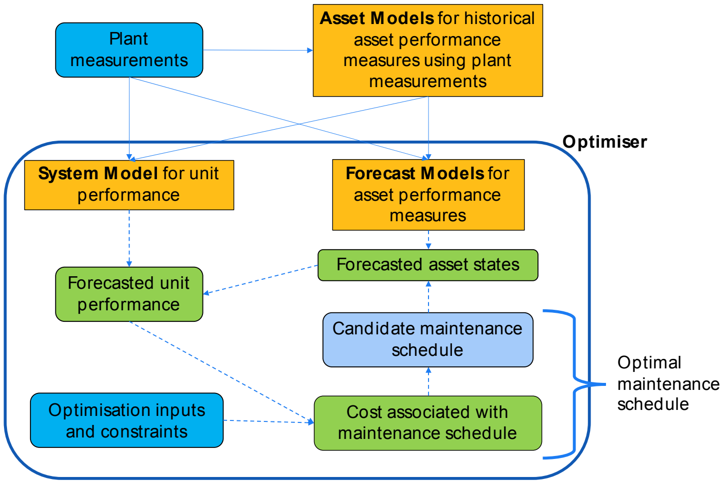

Figure 2: The Optimal Maintenance Scheduler Workflow

These are discussed in detail in the following sections.

Step 1: Asset models

The first step in the process involves the development of asset models for each asset in the unit that is to be considered for maintenance. An asset model takes measured tags as inputs and produces measures of asset performance as outputs.

There are various ways of creating asset models:

1. For certain types of assets, we can create physics-based models. For example, we can create physics-based models for shell and tube heat exchangers if their design sheets are available.
    - For common asset types, SymphonyAI Industrial has a **Physics Model Library** containing models that can be directly used.
    - For other assets, customers can create their own custom physics-based models in the SymphonyAI Industrial Eureka platform.
2. For assets where physics-based models cannot be created, we create data-based models. This is done using SymphonyAI Industrial's **Reduced Order Modeling** library which contains a variety of algorithms that can be used to build asset models using historical asset data.
3. In certain cases, it is possible to build so-called "hybrid" models that involve an underlying physics model which needs to be calibrated for the asset in question using historical asset data. These are particularly useful when there is a large set of simulation data available for an asset type; the "hybrid" model serves to "tweak" the simulation data for the asset in question.

As an example, in the refinery preheat train use case, asset models for heat exchangers are created using physics models with the flow rates and temperatures of the hot and cold streams as inputs and the overall heat transfer coefficient (or the fouling heat transfer coefficient, or the mean temperature difference) as the output.

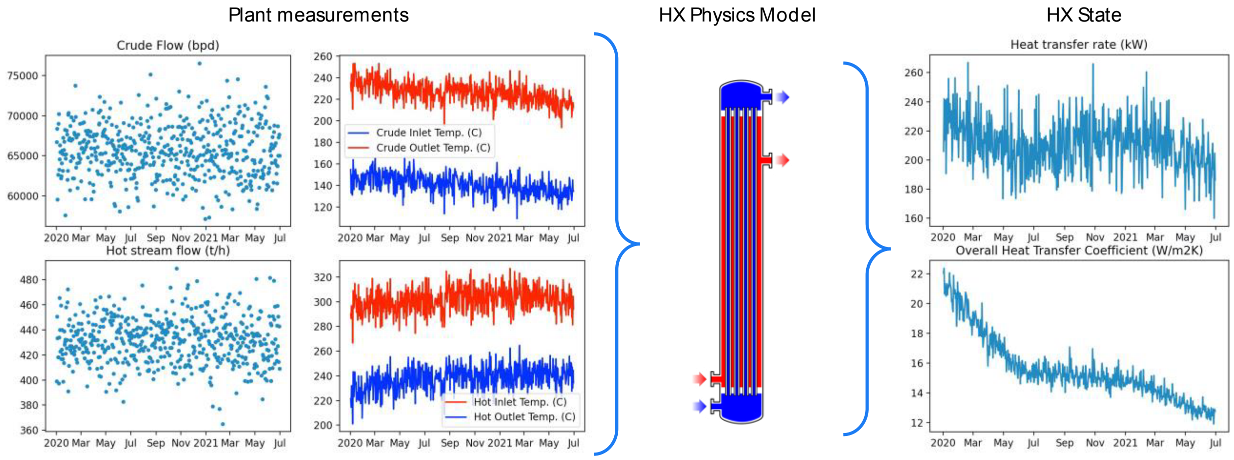

Figure 3: An example of a physics-based asset model used to determine asset performance measures

Step 2: System model for the unit

The next step in the process is the creation of a system model which calculates unit performance measures (e.g., throughput, inputs required, etc.) as functions of unit measurements and asset performance parameters (the latter are calculated from the asset models). Like the asset models, the system model can be physics-based, or data-based, or a hybrid.

1. For simple cases, a physics-based system model can be custom-defined in the SymphonyAI Industrial Eureka platform.
2. Alternately, where system models are too complex to define from scratch, customers may use their own process modeling software to generate simulation data points which can then be used with the Reduced Order Modeling library to create a system model. Hybrid models may also be possible.
3. In the absence of a simple physics-based model or process modeling software, historical system data may be used to create a data-based model using the Reduced Order Modeling library.

In the refinery preheat train use case, the system model is set up to calculate the furnace fuel required to heat the crude oil to the desired crude distillation unit (CDU) inlet temperature. The system model takes as inputs various unit measurements such as flow rates and temperatures.

$$\dot{m}_{furnace\ fuel} = f(\dot{m}_{crude}, T_{crude,in}, \dot{m}_{hot-1}, T_{hot-1,in}, \ldots, f_{HX-1}, f_{HX-2}, \ldots)$$

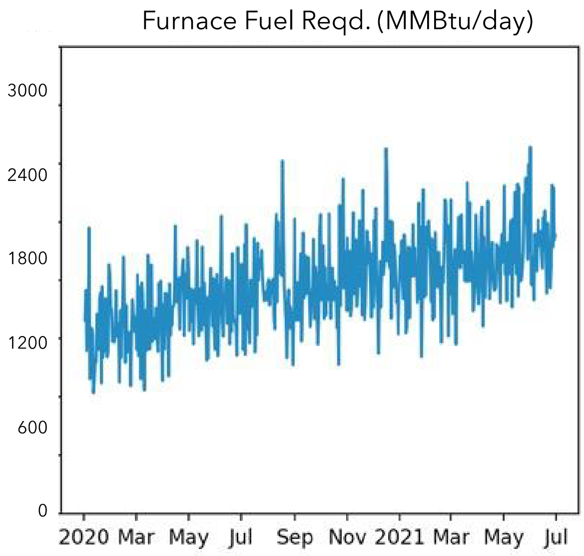

Figure 4: An example of a system model used to calculate the unit performance measure — in this case, the fuel required to maintain the crude oil temperature at a desired level

Step 3: Asset forecast models

The asset models developed in Step 1 are used along with historical measurements to develop asset **forecast** models which forecast the measures of asset performance into the future. These forecast models can be developed in one of several ways using SymphonyAI Industrial's **Deep Prophecy Engine**, a suite of forecasting algorithms:

1. We may forecast the asset model inputs into the future, then use those forecasted inputs with the asset model to forecast the asset model outputs (i.e., the measures of asset performance)
2. We may directly forecast the measures of asset performance.

In the refinery preheat train example, the heat exchanger overall heat transfer coefficients are forecast about 6-12 months into the future.

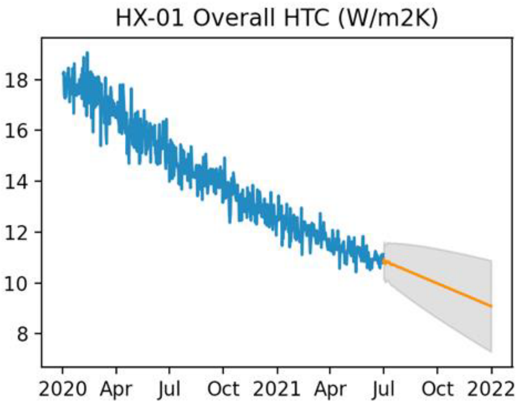

Figure 5: An example of a forecasting model used to forecast asset performance

Step 4: Optimisation

All the models required to run the maintenance optimiser are now in place. All that remains is for the optimisation inputs to be defined. Once these are defined, the optimiser determines which assets should undergo maintenance at what time.

Cost objective function

The cost objective function is evaluated over a specified time horizon (3 months, 6 months, etc.) and consists of two components:

1. The cost/revenue associated with the system throughput and/or system input consumables. This can in turn consist of several elements. Some examples include…

- The revenue associated with the throughput, based either on a fixed amount or on a time-varying amount over the specified time horizon.

$$Cost = -Revenue\ per\ unit \times Total\ throughput\ over\ the\ time\ horizon$$

$$Cost = -\int_{Time\ horizon} Rev.\ per\ unit(t) \times Throughput(t)dt$$

- A penalty associated with the throughput falling below a target level.

$$Cost = \int_{Time\ horizon} Penalty \times \max(0, Target - Throughput(t))\ dt$$

- Costs of consumables such as fuel or power.

$$Cost = \int_{Time\ horizon} Fuel\ cost(t) \times Fuel\ reqd.\ (t)dt$$

2. The cost associated with maintenance of assets (labour, material, etc.)

$$Cost = \sum_{Assets\ undergoing\ maintenance} Cost\ of\ maintenance\ of\ Asset\ i$$

The various costs and revenues required to evaluate the cost objective function are specified as **optimisation inputs**.

Constraints

It is possible to specify constraints on the maintenance schedule, e.g.:

1. Putting a limit on the number of assets that can concurrently undergo maintenance at any time, perhaps due to resource limitations.
2. Limiting maintenance activities to specified windows of time, perhaps to synchronise them with previously planned maintenance.

Optimisation process

The optimiser uses a Branch and Bound algorithm to determine which assets should undergo maintenance and at what times, to minimise the cost objective function. The optimisation process is represented in Figure 2. **It automatically accounts for reduced throughput during maintenance periods (due to the asset undergoing maintenance being offline) — as well as improved asset performance and therefore system performance following the maintenance**. For example, Figure 6 shows the evaluation of a scenario in which heat exchanger HX-03 is scheduled for maintenance after 30 days and HX-04 after 90 days. During the maintenance periods, the respective heat exchangers are offline, resulting in increased fuel requirements. However, after the maintenance events, the fuel requirements drop sharply due to improved asset performance.

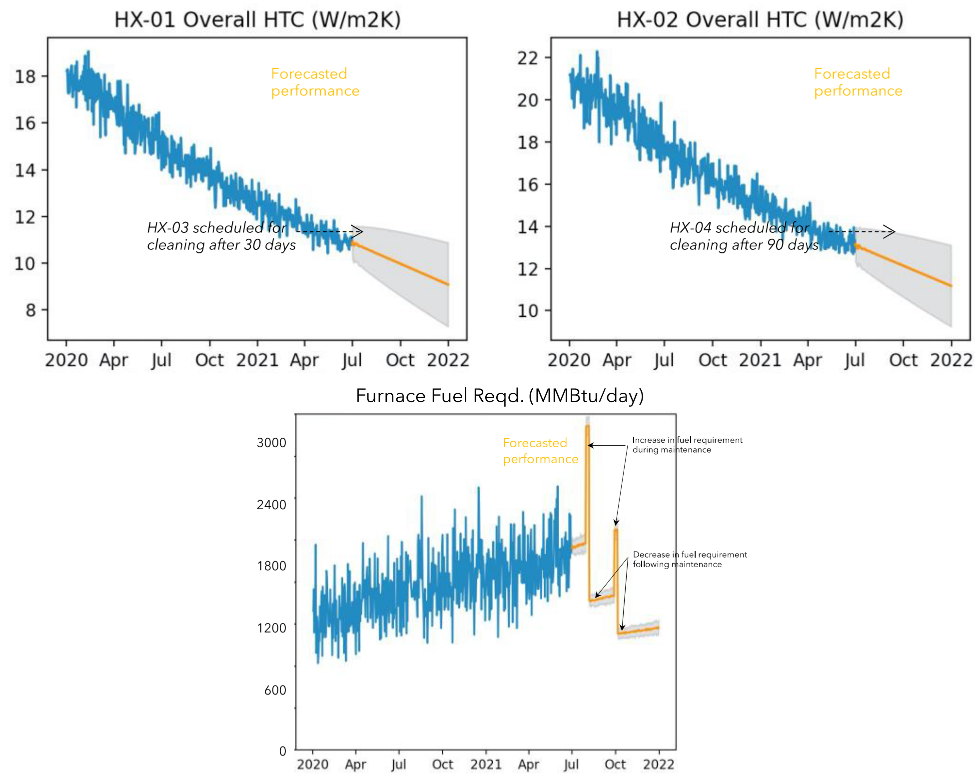

Figure 6: An example of how the optimiser automatically accounts for reduced throughput (or increased inputs) during maintenance periods

Solution

The optimiser provides at a minimum, the following outputs:

1. A maintenance schedule which includes a list of assets that should undergo maintenance as well as when they should undergo maintenance.
2. The benefit associated the optimal maintenance schedule relative to the scenario where there is no maintenance.

It can also be set up to provide a list of multiple maintenance schedules, each with its benefit analysis. This allows us the opportunity to choose from one of several maintenance schedules of similar benefit, based on any intangible criteria unaccounted for by the optimiser.

We may also choose to evaluate different sets of optimisation inputs. For example, we can run the optimiser with different cost/revenue scenarios and compare results from these scenarios. This comparison affords us the ability to hedge our bets with respect to maintenance schedules, i.e., we may choose to follow a schedule that is not necessarily optimal for a specific set of optimisation inputs but which provides consistent benefits across various scenarios.

Summary

SymphonyAI Industrial's Optimal Maintenance Scheduler is an easy-to-implement, highly customisable tool to plan maintenance activities for assets in a process unit in a way that maximises revenue or minimises cost at the plant level; it also allows for the comparison of different future scenarios. Its constituent models can either be built upon elements in SymphonyAI Industrial's Analytics Library such as the **Physics Modeling Library**, the **Deep Prophecy Engine**, and the **Reduced Order Modeling** framework, or can be custom-built according to specific needs; any changes to the constituent models are automatically reflected in all subsequent runs of the optimiser without manual intervention, making model management hassle-free.

All these features ensure that the OMS can be used across a wide variety of industrial applications, while being convenient to set up, customise, update, and manage.